Sections of United States Patent No.: 5,821,926 to Arita, cited in the Office Action of February 10, 2005.

## A.    Column 10, line 62 – Column 12, line 63 of Arita

Column 10, line 62 – Column 12, line 63of Arita disclose:

FIG. 2 shows an example of data structure together with data values stored in the memory 3. Each of data 1, data 2 and so forth consists of a plurality of fields such as field 1, field 2 and so forth. Of these fields, for example, field 1 stores "Tokyo" or "Osaka", field 2 stores "store T1" or "store T2", field 3 stores "commodity A" or "commodity B", field 4 stores "foodstuff" or "cosmetic", field 5 stores "550" or "150" as "budget" (unit: one million yen) among sales data, and field 6 stores "526" or "162" as "result" (unit: one million yen) among sales data.

As many data values as the number of data exist by very data-composing field, and the number of fields and the sequence of field arrangement are identical through all the data. The following description will be made providing that each date consists of six fields (field 1 to field 6) and that a total of 20 data (data 1 to data 20) are present.

Referring to FIGS. 3 to 5, it will be described how to not only enable the user to readily instruct the computer on processing within a short period of time on the basis of the above data, in particular, on the basis of the latest data stored without the need of separately developing operation-related software but also facilitate maintenance.

First, the data stored in the memory 3 is displayed on the display unit 1 and a field is selected and directed with the use of the input unit 2 to thereby automatically set a button class corresponding to the field selected and directed. Alternatively, a categorical field is automatically identified from the data values stored in the memory 3 and a button class corresponding to the identified field is automatically set. The terminology "categorical field" used herein means a field of other than that of a numerical value such as amount or quantity.

In this embodiment, field 1, field 2, field 3 and field 4 are selected and directed with the use of the input unit 2.

Alternatively, without conducting such selection and direction, field 5 and field 6 are automatically identified as numerical value fields and the other fields, i.e., fields 1, 2, 3 and 4 are automatically identified as categorical fields. Then, as shown in FIG. 3(b), fields 5 and 6 are excluded to fall outside button generation objects and button classes 1, 2, 3 and 4 are automatically set corresponding to fields 1, 2, 3 and 4.

With respect to the button classes 1 to 4, the button class designations are entered with the use of the input unit 2 and stored. However, the designations may be stored in another method, or they may automatically be given by programming and stored. For example, as shown in FIG. 3 (b), the designation for button class 1 corresponding to field 1 is "district", the designation for button class 2 corresponding to field 2 is "store", the designation for button class 3 corresponding to field 3 is "commodity", and the designation for button class 4 corresponding to field 4 is "department".

When the button classes are hierarchically correlated, the button class designations are displayed on the display unit 1 and selected in the hierarchically descending sequence of button class with the use of the input unit 2. The selection sequence per se is referred to as the hierarchical level. That is, herein, when the button class designations displayed on the display unit 1 are selected in the descending sequence: "district" and "commodity" with the use of the input unit 2, "district" is the upper level and "commodity" the lower level. Contrarily, when selection is made in the descending sequence: "commodity" and "district", "commodity" is the upper level and "district" is the lower level. The "store" and "department" falling outside the selection are uncorrelated button classes.

Individual buttons belonging to each button class are automatically set on the basis of the data values stored in the field corresponding to the button class. FIG. 4 shows an example of display of individual buttons made for each of button class 1 and button class 3 on the display unit 1.

In this example, a unique data value per se is employed as the individual button designation. As many individual buttons as the number of unique designations are automatically set in each single group. Herein, field 1 contains data values such as "Tokyo", "Osaka" and "Aichi", so that individual buttons with these designations are automatically set as shown in FIG. 4(a). On the other hand, field 3 contains data values such as "commodity A", "commodity B" and "commodity C", so that individual buttons with these designations are automatically set as shown in FIG. 4(b).

Appendix 1

In this example, in addition to those shown in FIG. 4, individual buttons with the designations "Hokkaido", "Fukuoka", "Kanagawa", "Saitama" and "Chiba" are automatically set in button class 1 and individual buttons with the designations "commodity D", "commodity E", "commodity F", "commodity G" and "commodity H" in button class 3 (Not shown in FIG. 4). Further, these individual buttons may have different designations, and it is feasible. to assign a unique designation to a range of data values or an aggregate of a plurality of data values and to automatically set as many individual buttons as the number of assigned designations.

When a plurality of button classes are correlated, a button group of hierarchically lower button classes correlated with individual buttons selected by a hierarchically upper button class is automatically set and switching displayed in the following manner.

That is, a certain button class is correlated hierarchically below another button class, thereby automatically setting a plurality of button groups corresponding to the certain button class in accordance with the situation of individual button selection in the other button class and effecting switching display. When there is no correlation with other button classes, a single button group is automatically set corresponding to the certain button class.

For example, when "district" of button class 1 is set hierarchically low and "commodity" of other button class 3 is correlated hierarchically above the same, the selection of individual buttons "commodity A", "commodity B" and "commodity C" of the hierarchically upper button class 3 leads to automatic setting of button group 1-1 having individual buttons with the designations such as "Tokyo", "Osaka" and "Aichi" as shown in FIG. 5(a). When only individual button "commodity A" of the hierarchically upper button class 3 is selected, the data storing "commodity A" in field 3 are, for example, data 1, data 4 and data 7 of FIG. 2. "Aichi" does not exist in the respective fields 1 of these data, so that button group 1-2 having individual button "Aichi" removed therefrom is automatically set as shown in FIG. 5(b).

Likewise, when "commodity" of button class 3 is set hierarchically low and "district" of other button class 1 is correlated hierarchically above the same, the selection of individual buttons "Tokyo", "Osaka" and "Aichi" of the hierarchically upper button class 1 leads to automatic setting of button group 3-1 having individual buttons with the designations such as "commodity A", "commodity B" and "commodity C" as shown in FIG. 5(c). When only individual

3                                                    Appendix 1

button "Aichi" of the hierarchically upper button class 1 is selected, the data containing "Aichi" in field 1 is only data 6 of FIG. 2. Thus, button group 3-2 having individual button "commodity C" generated on the basis of the data value stored in field 3 of data 6 is automatically set as shown in FIG. 5(d).

**B.     Column 20, line 32 – Column 21, line 65 of Arita**

FIG. 15 shows a link table which is the most suitable for use in the above situation. In this link table, a unique button selection pattern is used as an index and as many data Nos. corresponding to individual button selection patterns as existing are held in the same index. In this case, arranging button selection patterns in a given sequence prior to use enables more efficiently identifying corresponding data at a greater speed, for example, by searching for the button selection pattern conforming to the position of button selection by binary search.

FIGS. 16 to 21 show an example in which, when the same mutually correlated button manipulations as before are needed, repetition of the manipulations can partially or entirely be avoided depending of the user's decision at that time. FIGS. 16(a) and (b) respectively show button classes "report partition" and "store" conducting selection and direction for data retrieval in computer processing. FIG. 16(c) shows button class "management item" conducting selection and direction for executing processing based on the retrieved data.

That is, one button group "business results" is set for the button class "report partition" shown in FIG. 16 (a), and the button group "business results" is indicated as an aggregate of individual buttons such as those with the designations "daily", "weekly", "monthly" and "annual". The button class "store" shown in FIG. 16(b) is the same as that shown in FIG. 7(c). Further, one button group "summary" is set for the button class "management item" shown in FIG. 16(c), and the button group "summary" is indicated as an aggregate of individual buttons such as those with the designations "sales", "gross profit", "ratio of sales to that of the same month of the preceding year", "ratio of gross profit to that of the same month of the preceding year", "sales performance" and "gross profit performance". In this example, with respect to the selection of individual buttons, the selection among button classes is conducted in the sequence: "report partition", "store" and "management item". That is, the following description will be made providing that the individual button is selected in the button class sequence: "report partition", "store" and

"management item". Each of FIGS. 17 to 19 shows the operating button selection sequence during a series of button manipulations effected by displaying the operating buttons shown in FIG. 16 on the display unit 1 and sequentially selecting individual buttons and examples of selected individual buttons.

In this example, referring to FIG. 17, the selection sequence 1 selects individual button "daily" of the button group "business results" of the button class "report partition"; the selection sequence 2 selects individual buttons "store T1" and "store T2" of the button group "Kanto district" of the button class "store"; and the selection sequence 3 selects individual buttons "sales" and "gross profit" of the button group "summary" of the button class "management item". Likewise, referring to FIG. 18, the selection sequence 1 selects individual button "monthly"; the selection sequence 2 selects individual button "major store"; and the selection sequence 3 selects individual buttons "sales", "gross profit", "ratio of sales to that of the same month of the preceding year", "ratio of gross profit to that of the same month of the preceding year", "sales performance" and "gross profit performance". Further, referring to FIG. 19, the selection sequence 1 selects individual button "monthly"; the selection sequence 2 selects individual button "new store"; and the selection sequence 3 selects individual buttons "sales", "gross profit", "sales performance" and "gross profit performance".

The state of storing a manipulation history in a button manipulation history table set in the memory 3 on the basis of the above individual button selecting manipulations shown in FIGS. 17 to 19 will now be described with reference to FIG. 20.

This button manipulation history table consists of a column of history numbers provided with consecutive numbers, a manipulation history column on which the manipulation history is sequentially written and stored and a latest position column in which a latest position mark, e.g., ".diamond-solid." is provided. In this table, the history No. 1 corresponds to a sequence of button manipulations shown in FIG. 17; the history No. 2 corresponds to a sequence of button manipulations shown in FIG. 18; and the history No. 3 corresponds to a sequence of button manipulations shown in FIG. 19.

That is, the above each manipulation history is one obtained by selecting individual buttons of the button classes "report partition", "store" and "management item" in this order and indicating the selection sequence among the button classes and the position of selection in each button class while

correlating them. The sign of inequality (>) delimits button classes and the digit indicates the position of selection of individual button counted from top in the sequence: 01, 02, . . . . When a plurality of individual buttons are selected in a single button class, the selection positions are delimited by commas (,) and arranged.

For example, the manipulation history of the history No. 1 shown in FIG. 20 corresponds to a series of button manipulations shown in FIG. 17 and indicates sequential manipulations of selecting first individual button "daily" (01) of the button class "report partition", secondly individual buttons "store T1" (03) and "store T3" (04) of the button class "store" and thereafter individual buttons "sales" (01) and "gross profit" (02) of the button class "management item".

## C.   Column 24, line 44 – Column 25, line 21 of Arita

FIG. 24 shows an example in which the display sequence and display sizes of individual buttons among the same button group of the same button class are changed according to the data values stored in the memory 3 corresponding to the individual buttons and displayed.

That is, in this example, the data value, sales here, corresponding to each individual button of the button group "whole country" of the button class "district" as shown in FIG. 22(a) is secured, and the display sequence and display size of each individual button are changed on the basis of the data value (sales) and displayed. An "individual button/corresponding data value table" as shown in FIG. 24(a) is set and stored in the memory 3.

In this example, the sales column of the "individual button/corresponding data value table" exhibits the sum by every individual button of the values as sales of field 6 of the data (data 1 to 7) shown in FIG. 2.

Illustratively, the sales "868" corresponding to individual button "Tokyo" represent the sum of values of field 6 "526", "162" and "180" of data 1, 2 and 3, respectively, whose field 1 recitations are "Tokyo". The sales "124" corresponding to individual button "Osaka" represent the sum of values of field 6 of data 4 and 5, respectively, whose field 1 recitations are "Osaka". With respect to each of individual buttons "Aichi" and "Hokkaido" as well, the sales represent the sum of values of field 6 of the data whose field 1 recitations are the concerned district.

Referring to FIG. 24(b), the display sequence of individual buttons can be rearranged in the order of sales

amount and redisplayed by directing "change of display sequence according to sales" with the use of the input unit 2 by means of, for example, a button. Thus, "Tokyo" whose sales are the greatest can be displayed at the forefront (topmost), thereby the user can find the individual button whose sales are great easily.

Further, a "change of display size according to sales" is directed by means of, for example, a button with the use of the input unit 2 so that the display sizes of individual buttons are redisplayed in proportion to sales as shown in FIG. 24(c). Thus, individual button "Tokyo" whose sales are the greatest can be displayed at the forefront (topmost) in the largest size, thereby enabling the user to more quickly find individual button whose sales are great.

As shown in FIGS. 23(d) and (e), the direction sequence can be reversed and, naturally, only either the display sequence or the display size can be changed.

## D.    Column 15, line 13 – Column 16, line 54 of Arita

That is, FIG. 8(a) shows the state of arrangement exhibited after multiplication of the new individual buttons "major store" and "new store" as shown in FIG. 7 (c). In this state, selection of the individual button "major store" (new button) as shown in FIG. 8(b) results in emulation of the manipulation of selecting two buttons "store T1" and "store T3" in the memory 3 as shown in FIG. 8(c). Likewise, selection of the individual button "new store" (new button) results in emulation of the manipulation of selecting three buttons "store K1", "store S2" and "store C1" in the memory 3.

In the fetching of data or information under the various conditions, if the various needs are integrated under a direct designation on the basis of the existing individual buttons and a new summarizing individual button is defined and multiplied, the computer 4 can be easily instructed on the execution of processing by selection of the new individual button, thereby avoiding the need of conducting a time consuming manipulation of selecting all the series of existing individual buttons as objects of needs.

In this example, for example, when it is assumed that there are needs of fetching data or information on stores limited to those especially important with respect to sales and profit, it is satisfactory to select one new individual button "major store" integrating the stores instead of the selection of the above defined two existing individual buttons "store T1" and "store T3" as major stores. This also applies when it is

assumed that there are needs of fetching data or information on sales and profit limited to those of the new stores, it is satisfactory to select one new individual button "new store" integrating the stores instead of the selection of three existing individual buttons "store K1", "store S2" and "store C1".

The above definition of a new individual button on the basis of the combination of existing individual button selections and multiplication thereof avoids the need of fully memorizing a plurality of individual button designations regarding certain conditions and making selection. Thus, the greater the number of existing individual buttons as objects, the less the manipulation labor. Further, the integration of needs under a direct designation accompanied by summarization leads to easy understanding with the result that selection errors and omissions can be prevented.

After the definition and multiplication of new individual buttons by the above method, these individual buttons can be regarded as existing individual buttons and included in the combination of existing individual button selections to thereby enable definition and multiplication of other new individual buttons. In this manner, according to necessity, new individual buttons can successively be multiplied to thereby attain further simplification of button manipulation.

FIG. 9 shows an example of definition and multiplication of a new individual button on the basis of existing individual buttons by correlating selection combinations with respect to two button classes, i.e., the above button class "store" and another button class "commodity". Herein, the button class "commodity" corresponds to field 3 of data shown in FIG. 2 as mentioned above. In this example, one button group "major classification" is set for the button class "commodity", which is expressed as an aggregate of individual buttons such as those with the designations "commodity A" and "commodity B". In this example as well, referring to FIG. 9(b), the definition of a new individual button is conducted by selecting as many arbitrary individual buttons as required from those existing buttons with the use of the input unit 2 and thereafter entering the designation of the new individual button integrating these with the use of the input unit 2.

In this example, two individual buttons "store T1" and "store T3" are selected from the existing individual buttons of the button group "Kanto district" of the button class "store" with the use of the input unit 2. Then, individual buttons "commodity A" and "commodity B" are similarly selected from the existing individual buttons of the button group "major classification" of the button class "commodity". A new individual button with the

designation "selective management object" is defined by correlating selection combinations with respect to the above two button classes.

In the multiplication of the thus defined new individual button, referring to FIG. 9(c), a button class "management item" and a button group "Kanto district" suitable for the individual button "selective management object" are newly set, wherein the individual button (new button) is displayed as an operating button. Selection of the new individual button "selective management object" leads to emulation of the manipulation of selecting four buttons, i.e., the above defined "store T1" and "store T3" of the button class "store", and "commodity A" and "commodity B" of the button class "commodity".

## E.   Abstract of Arita

Method of generating an operating button for computer processing, method of retrieving data with the operating button and method of displaying the operating button.

Appendix 1